



US 20060197982A1

(19) **United States**

(12) **Patent Application Publication**

Bell et al.

(10) **Pub. No.: US 2006/0197982 A1**

(43) **Pub. Date:**

Sep. 7, 2006

(54) **DESIGNER-CREATED ASPECT FOR AN ELECTRONIC FORM TEMPLATE**

Publication Classification

(75) Inventors: **Joshua S. Bell**, Kirkland, WA (US);
Scott M. Roberts, Bothell, WA (US);
Jun Jin, Bellevue, WA (US); **Brian C. Teutsch**, Seattle, WA (US); **Laurent Mollicone**, Kirkland, WA (US)

(51) **Int. Cl.**
G06K 15/00 (2006.01)
(52) **U.S. Cl.** **358/1.18**

(57) **ABSTRACT**

Systems and/or methods that enable addition of a designer-created aspect originating from one electronic form template into another electronic form template are described. The systems and/or methods may also, in one embodiment, enable a user to graphically package designer-created aspects of an existing electronic form template into a component capable of being added to another electronic form template. In another embodiment, these systems and/or methods may enable mapping of an existing non-structural designer-created aspect originating from one electronic form template onto a data structure of another electronic form template.

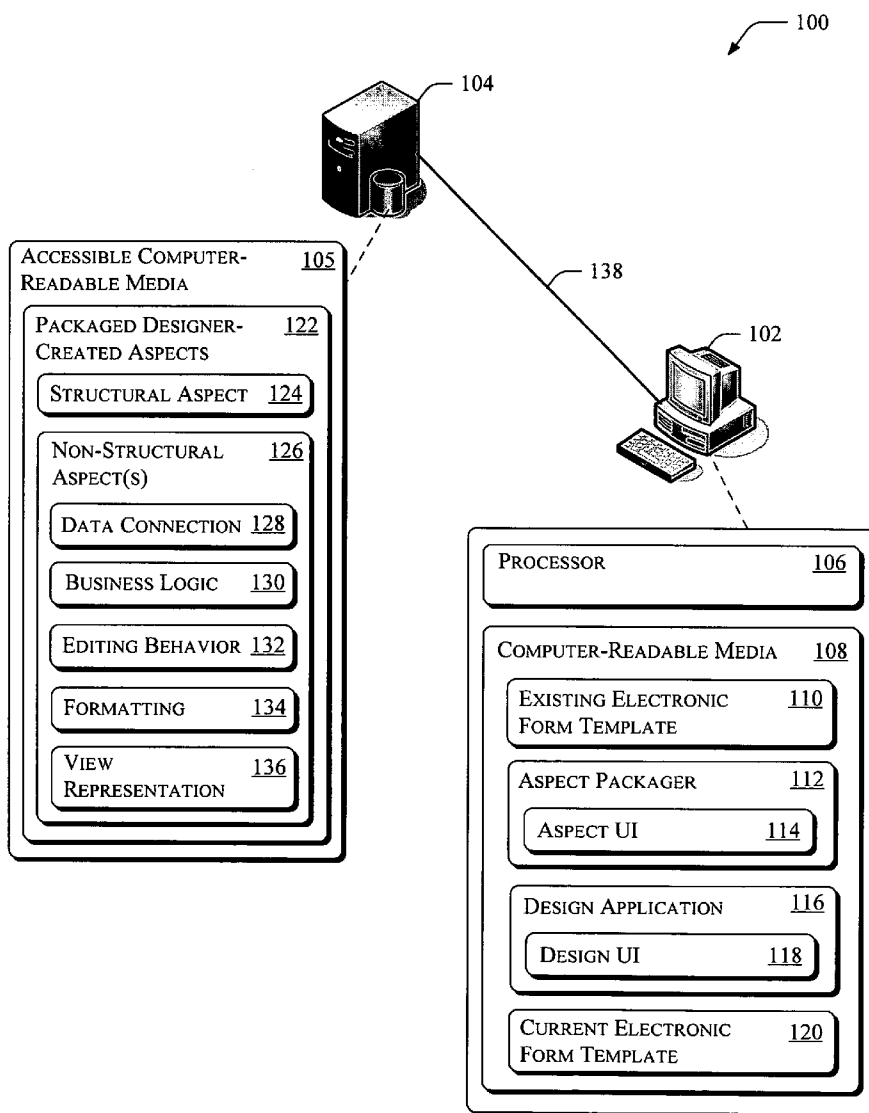
Correspondence Address:

LEE & HAYES PLLC
421 W RIVERSIDE AVENUE SUITE 500
SPOKANE, WA 99201

(73) Assignee: **Microsoft Corporation**, Redmond, WA

(21) Appl. No.: **11/072,087**

(22) Filed: **Mar. 4, 2005**



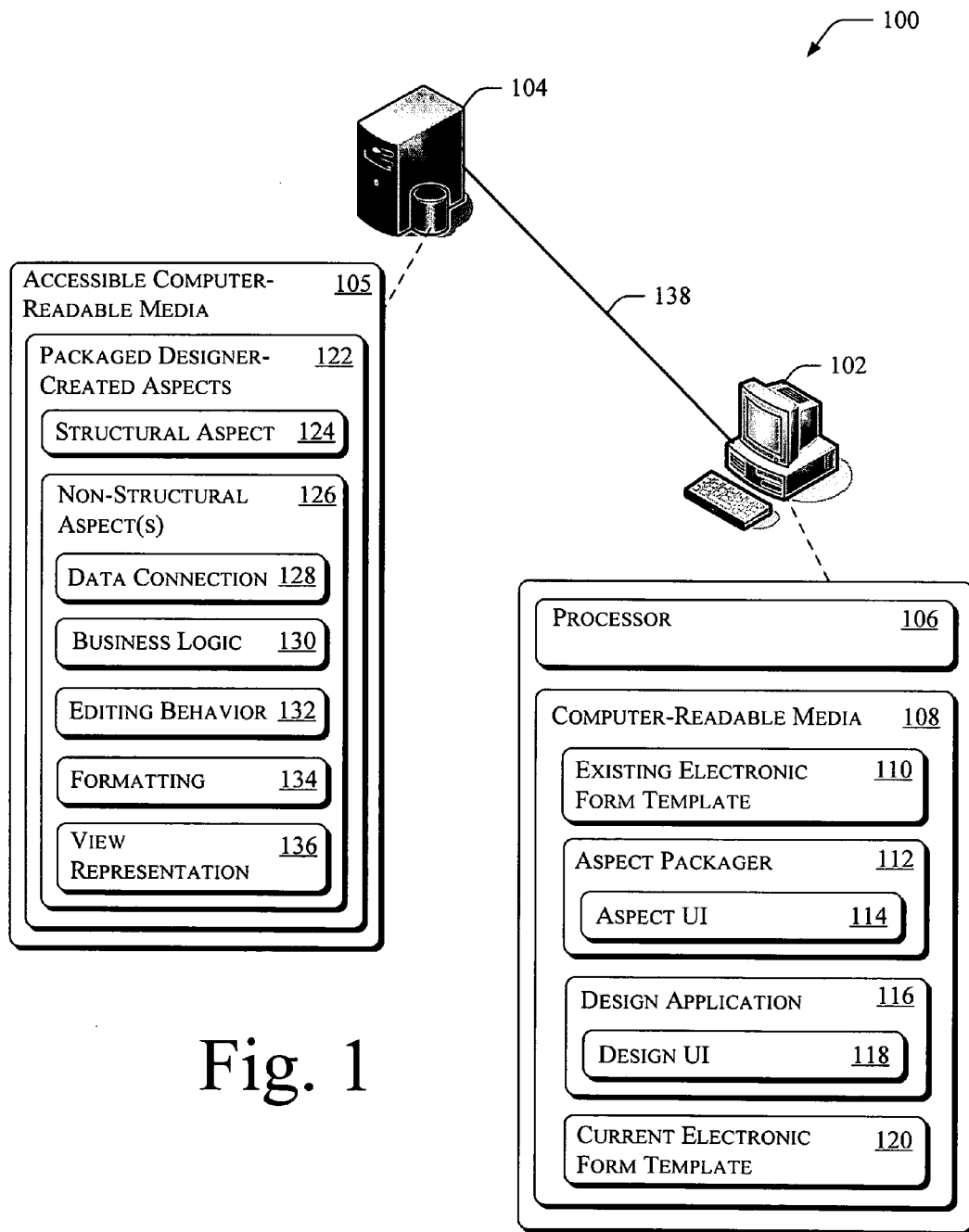


Fig. 1

Rendered View of Existing Electronic Form Template

200 **TIRE SALES RECORD**

Acme Tire Company
A Family-Owned Company

202	Salesman Name:	206	Type of Tire:
204	Employee No.:	208	Tire Cost:
		210	Total Price:

224

Customer Information

212	Car Type:		
214	Customer Name:		
216	Street:		
218	City:	222	ZIP Code:
220	State:		

Fig. 2

300 →

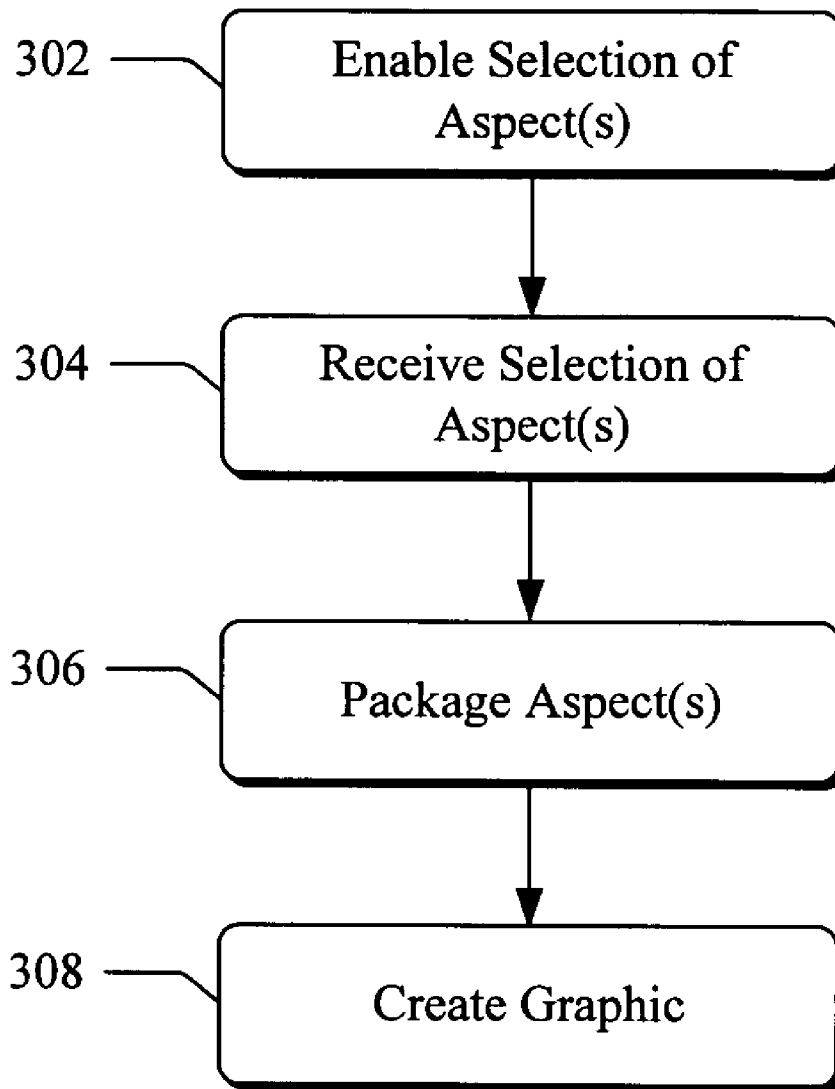


Fig. 3

200

The image shows a screenshot of a web browser window titled "Rendered View of Existing Electronic Form Template". The browser's address bar contains the text "Type a question for help". The main content of the page is a form titled "TIRE SALES RECORD" for "Acme Tire Company", which is described as a "Family-Owned Company". The form is divided into several sections:

- A top section with two columns of input fields:
 - Left column: "Salesman Name:" and "Employee No.:"
 - Right column: "Type of Tire:", "Tire Cost:", and "Total Price:"
- A larger section titled "Customer Information" containing:
 - "Car Type:"
 - "Customer Name:"
 - "Street:"
 - "City:"
 - "State:" and "ZIP Code:" (split across two columns)

The form fields are enclosed in dashed-line boxes. A reference numeral "402" points to the "Customer Information" section.

Fig. 4

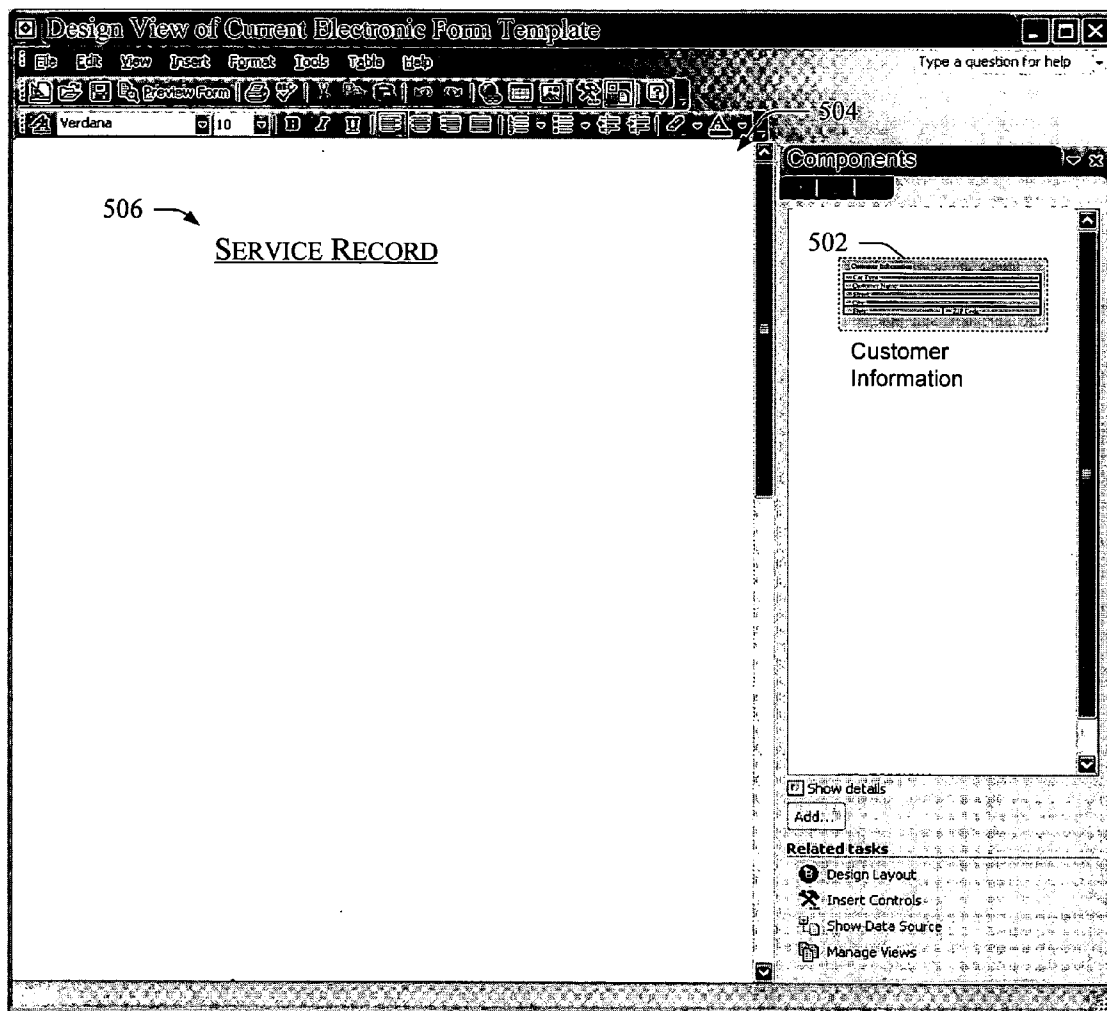


Fig. 5

600 →

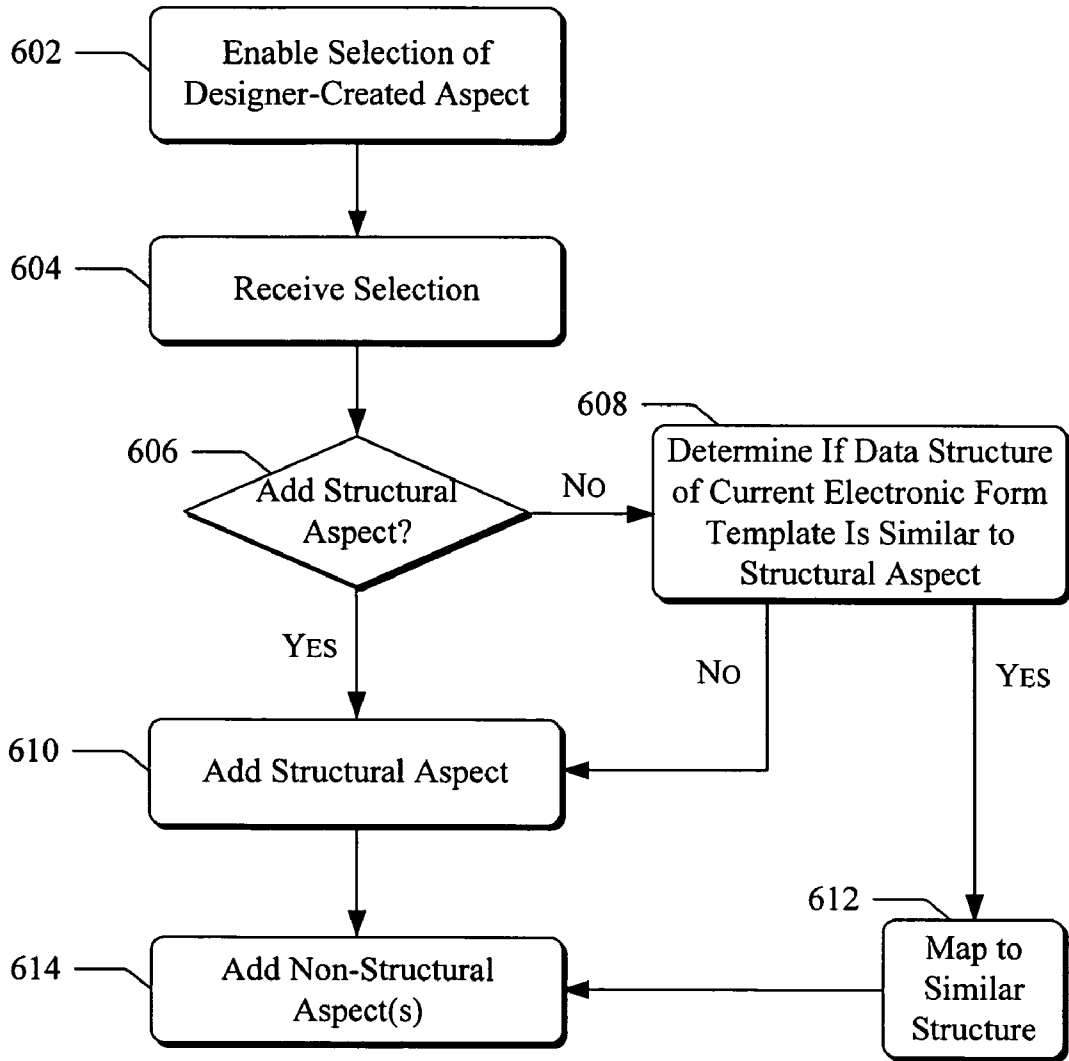


Fig. 6

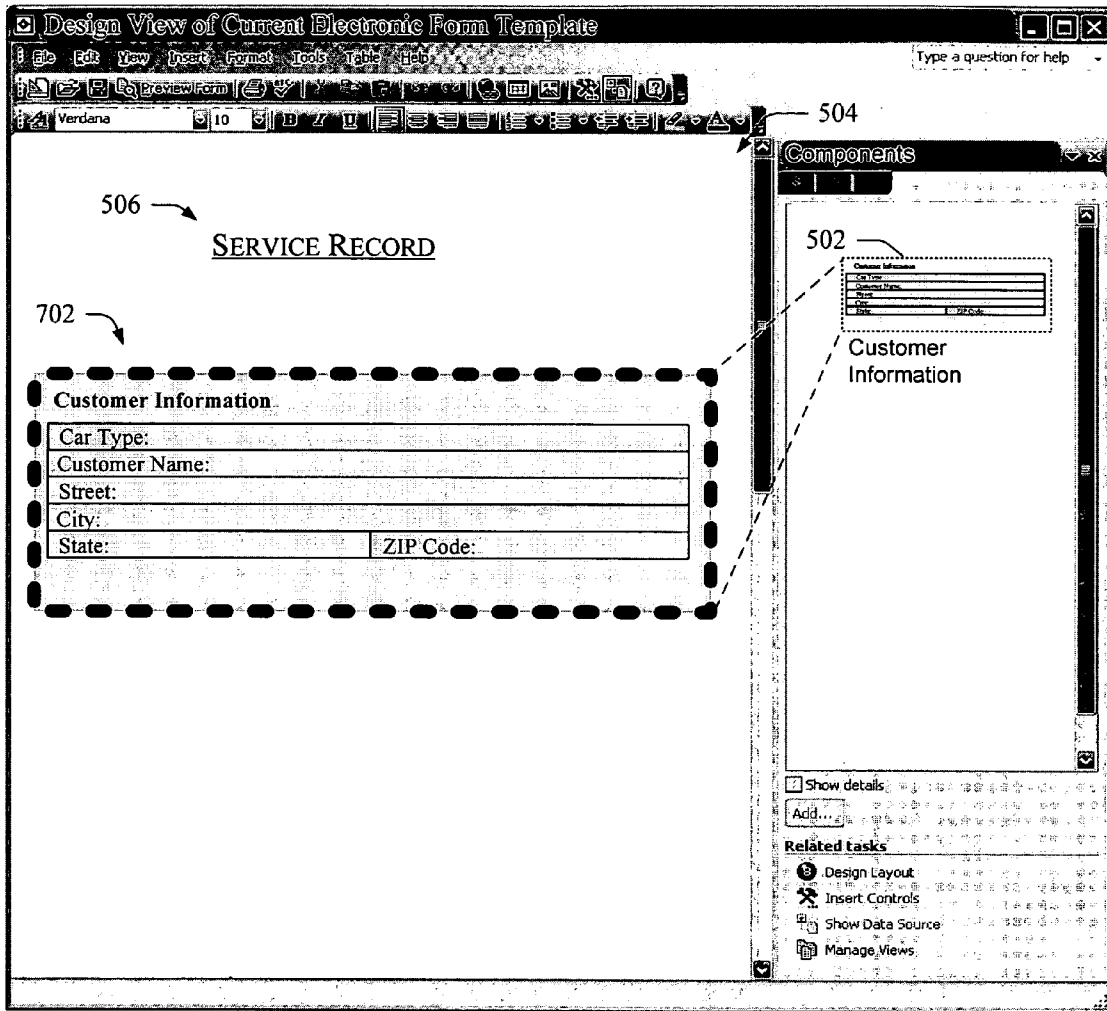


Fig. 7

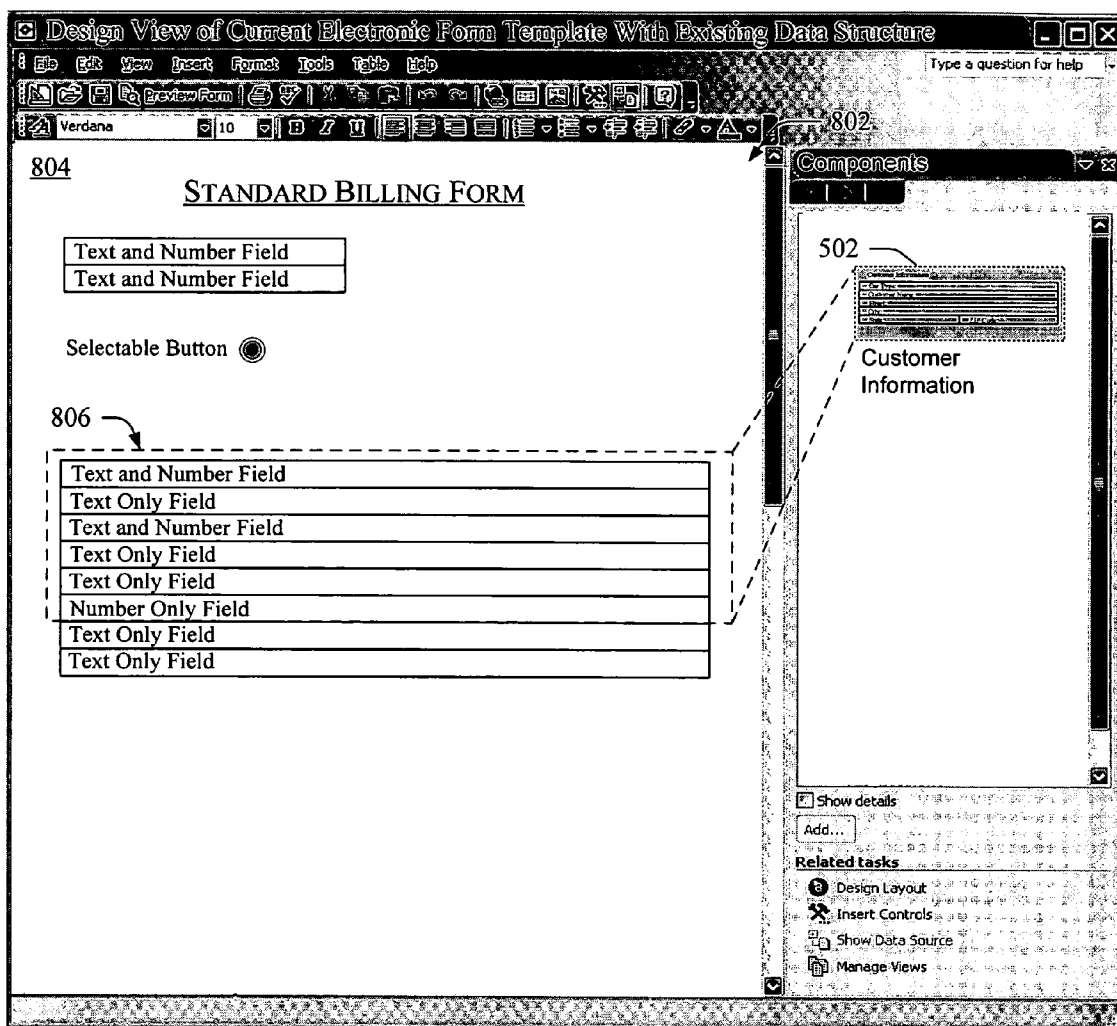


Fig. 8

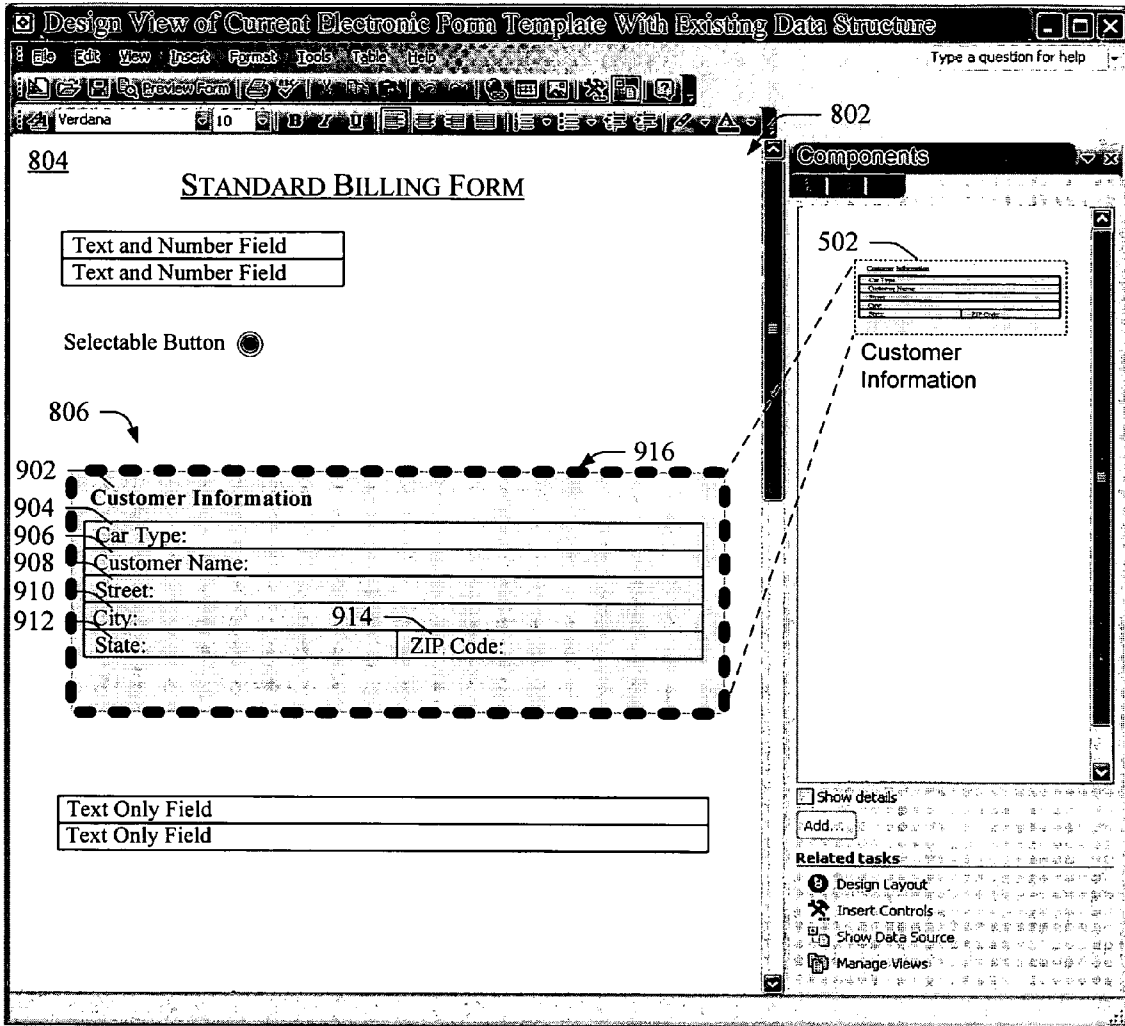


Fig. 9

DESIGNER-CREATED ASPECT FOR AN ELECTRONIC FORM TEMPLATE

TECHNICAL FIELD

[0001] This invention relates to electronic form templates.

BACKGROUND

[0002] Electronic forms are commonly used to collect information. These electronic forms may enable users to enter data and have that data stored digitally, such as in computer-accessible databases. Data so stored can be quickly retrieved, allowing others to use that data.

[0003] Electronic form templates may be designed to collect specific kinds of information or to collect it in particular ways. An electronic form template for recording tire sales, for instance, may be designed to enable entry of a tire salesman's name and employee number, the type of tire sold, the cost of the tires, and customer information. With this recorded information, the tire company may better be able to keep track of its sales, inventory, and customer information.

[0004] For consistency across multiple forms or to save time and effort, a programmer may wish to reuse part of an electronic form template when building another electronic form template. A programmer may, for example, want to reuse the part of the tire sales record for recording a customer's information in another electronic form template needing entry of such information. To do so, however, the programmer may have to write code, often requiring substantial time or programming experience.

SUMMARY

[0005] Systems and/or methods ("tools") are described that enable addition of a designer-created aspect originating from one electronic form template into another electronic form template.

[0006] In one embodiment, the tools enable—without programming—a user to package designer-created aspects of an existing electronic form template into a component capable of being added to another electronic form template.

[0007] In another embodiment, the tools enable mapping of a non-structural designer-created aspect onto a data structure of an electronic form template.

BRIEF DESCRIPTION OF THE DRAWINGS

[0008] FIG. 1 illustrates an exemplary operating environment.

[0009] FIG. 2 illustrates an exemplary rendered view of an existing electronic form template.

[0010] FIG. 3 is a flow diagram of an exemplary process for enabling a designer to package designer-created aspects.

[0011] FIG. 4 illustrates the view of FIG. 2 with a selection box.

[0012] FIG. 5 illustrates a design view of an exemplary current electronic form template.

[0013] FIG. 6 is a flow diagram of an exemplary process for enabling a designer to add a designer-created aspect to an electronic form template.

[0014] FIG. 7 illustrates the view of FIG. 5 with addition of exemplary, packaged designer-created aspects.

[0015] FIG. 8 illustrates a design view of an exemplary current electronic form template having an existing data structure.

[0016] FIG. 9 illustrates the view of FIG. 8 with addition of exemplary non-structural aspects.

[0017] The same numbers are used throughout the disclosure and figures to reference like components and features.

DETAILED DESCRIPTION

Overview

[0018] The following disclosure describes a user-friendly way to build into a current electronic form template aspects that previously have been created, such as those originating from an existing electronic form template. A previously created aspect may be built into a current electronic form template without requiring that a form designer have programming experience.

[0019] In one embodiment, a user may, without programming, package designer-created aspects of an existing electronic form template into a component that is capable of later being built into another electronic form template. These aspects may comprise various customizations made to the existing electronic form template, such as a structural aspect and non-structural aspects like data connections, business logic (e.g., validation and rules governing data), editing behavior, formatting, and view representations.

[0020] Aspects added to a current electronic form template may be non-structural and mapped to an existing data structure of the current electronic form template. This permits, for example, addition of previously existing, designer-created non-structural aspects to an electronic form template having a data structure that may not be changed.

Exemplary Operating Environment

[0021] Before describing the tools in detail, the following discussion of an exemplary operating environment is provided to assist the reader in understanding where and how the tools may be employed. The description provided below constitutes but one example and is not intended to limit application of the tools to any one particular operating environment.

[0022] FIG. 1 illustrates one such operating environment generally at 100 comprising a computer 102 and an accessible data source 104 having accessible computer-readable media 105. The computer is shown comprising a processor 106 and computer-readable media 108. The processor is capable of accessing and/or executing the computer-readable media. The computer-readable media comprises an existing electronic form template 110, an aspect packager 112 having an aspect user interface 114, a design application 116 having a design user interface 118, and a current electronic form template 120. The aspect packager and the design application are shown separate but may be combined.

[0023] The aspect packager is capable of enabling a user to package one or more existing designer-created aspects (shown packaged at 122) for later addition to another electronic form template, such as current electronic form template 120. Packaged designer-created aspects 122 com-

prise a structural aspect **124** and non-structural aspects **126**. The non-structural aspects may comprise various customizations made to a form template, like a data connection aspect **128**, a business logic aspect **130**, an editing behavior aspect **132**, a formatting aspect **134**, and a view representation aspect **136**.

[0024] The design application is capable of enabling addition of the packaged designer-created aspects to the current electronic form template. These packaged designer-created aspects are accessible by the computer, such as by being stored on the computer's memory or across a network **138** at the accessible data source **104** (as shown).

An Exemplary Existing Electronic Form Template

[0025] An exemplary existing electronic form template is described below. This example form template is provided to aid the reader in understanding various designer-created aspects of electronic form templates that may be packaged for later reuse.

[0026] A rendered view **200** of existing electronic form template **110** is shown in **FIG. 2**. The view shows an electronic form template for entering a sale of tires with data-entry fields for a tire salesman's name **202**, an employee number **204**, a type of tire **206**, a tire cost **208**, a total price **210**, a car type **212**, a customer name **214**, a customer street **216**, a customer city **218**, a customer State **220**, and a customer ZIP code **222**.

[0027] These data-entry fields and their arrangement represent an existing, designer-created structural aspect for the existing electronic form template. The data-entry fields also may have different properties that affect their structure, such as by being repeatable (e.g., the type of tire sold and tire cost, like if the customer purchased one type of his front axel and another for his rear axel), optional (e.g., the tire salesman's name, like if his employee number indicates his name and so entering his name is redundant), only permitting entry of text (e.g., the customer's name data-entry field), or only permitting entry of numbers (e.g., the tire cost, total price, and ZIP code fields). The designer of this illustrated existing electronic form template customized it by selecting these particular types of data-entry fields, where they are arranged in the view, and where they reside within the existing electronic form template's data structure.

[0028] The existing electronic form template has other (non-structural) aspects created by its designer. Some of these are visual, and so are easy to see in this rendered view. One such aspect comprises formatting. The formatting may determine the font, size, and color of the text in and around the data-entry fields, such as the text "TIRE SALES RECORD", "Acme Tire Company", "A Family-Owned Company"; "Salesman Name:;", "Employee No.:". "Type of Tire:;", "Tire Cost:;", "Total Price:;", "Customer Information", "Car Type:;", "Customer Name:;", "Street:;", "City:;", "State:;", and "ZIP Code:;". Another visual aspect is the view representation. This aspect may determine the color and size of the data-entry fields. Data-entry fields **212**, **214**, **216**, **218**, **220**, **222**, and the text "Customer Information", for instance, are within a shaded, dashed-line box marked at **224**. This shaded, dashed-line box is one view representation aspect of the existing electronic form template.

[0029] The existing electronic form template may comprise other non-structural aspects, such as aspects for data

connection, business logic, and editing behavior. One data connection aspect auto-populates the city and State fields. If the salesman enters a ZIP code into the customer zip code data-entry field, for instance, the existing electronic form template may connect to a data source and determine, based on this entered ZIP code, a corresponding city and State. With this information, the existing electronic form template may then populate this corresponding city and State into the city and State data-entry fields, thereby saving the salesman time and possibly reducing the chance of error.

[0030] Business logic aspects may perform operations on data entered into the fields. If a particular price is entered into tire cost data-entry field **208**, for instance, business logic (e.g., script or managed code) may operate to multiply this number by four (assuming four tires were purchased), add appropriate tax, and enter this total price into total price data-entry field **210**.

[0031] An aspect for editing behavior may govern how a user is permitted to interact with the existing electronic form template, such as by not permitting a salesman to enter more than a certain number of characters into the employee number field.

[0032] Each of these non-structural aspects may be linked to a structural aspect, in this case to corresponding data structures in the existing electronic form template. In one embodiment, each of these links is separable from its associated structure. This separability is described in greater detail below.

Packaging Aspects for Reuse

[0033] The following discussion describes exemplary ways in which the tools enable a user to package aspects for reuse.

[0034] An exemplary process **300** for enabling a user to package aspects, such as those created by a designer and originating from an existing electronic form template, is shown in **FIG. 3**. Process **300** is illustrated as a series of blocks representing individual operations or acts performed by elements of the operating environment **100** of **FIG. 1**, such as aspect packager **112** and aspect user interface **114**. This and other processes disclosed herein may be implemented in any suitable hardware, software, firmware, or combination thereof; in the case of software and firmware, these processes represent a set of operations implemented as computer-executable instructions stored in computer-readable media **108** and executable by processor **106**.

[0035] At block **302**, the aspect packager enables selection of designer-created aspects. A user is able, for instance, to select an existing designer-created aspect with a text-based command or graphically, such as by graphically selecting a portion of a rendered existing electronic form template representing designer-created aspects.

[0036] Continuing the illustrated embodiment, the aspect packager enables selection through the rendered view **200** of the existing electronic form template as shown in **FIG. 2**. Though this view, the aspect user interface enables a user to select all or particular designer-created aspects of the existing electronic form template for later reuse.

[0037] Assume, for example, that the designer of a form template wishes to reuse aspects of the customer information data-entry fields (fields **212**, **214**, **216**, **218**, **220**, and **222**).

The form designer may want to have these aspects available for reuse in other form templates, such as a service record for recording automotive services like fixing a car's brakes.

[0038] At block 304, the aspect packager receives the user's selection. As mentioned, this selection can be textual, graphical, and otherwise not require that the user program or write code. The selection may comprise all of the existing designer-created aspects of the existing electronic form template or portions of it.

[0039] Continuing the illustrated embodiment, the user selects the customer information data-entry fields by drawing a box around these fields, shown with a selection box 402 of FIG. 4.

[0040] At block 306, the aspect packager packages the designer-created aspects selected by the user effective to enable one or more of the aspects to be added to another electronic form template. These aspects may be packaged separately or as a group. In one embodiment, these aspects are combined in one, inseparable group. In other embodiments, this packaging is effective to enable the aspects to be added separately to an electronic form template, such as by enabling non-structural aspects to be mapped to an existing data structure of a current electronic form template that has a structure similar to that of the structural aspect associated with these non-structural aspects.

[0041] The aspects may be packaged separately and saved locally or at a remote source, such as the six aspects saved at remote source 104 shown in FIG. 1: structural aspect 124; data connection aspect 128; business logic aspect 130; editing behavior aspect 132; formatting aspect 134; and view representation aspect 136. The aspects may be packaged and saved using a markup language (e.g., eXtensible Markup Language, "XML"), a transformation language (e.g., XML style sheet language transformation), an extensible style sheet language (e.g., eXtensible Stylesheet Language), a schema (e.g., XML schema), or as hypertext machine language (HTML), for example.

[0042] The aspect packager may also package aspects into a component, the component comprising the structural aspect separable from the associated non-structural aspects (e.g., those shown comprised in non-structural aspects 126 of FIG. 1). This component may comprise the structural and non-structural aspects packaged effective to enable the non-structural aspects to be added to an electronic form template without having to also add the structural aspect.

[0043] Continuing the illustrated embodiment, the aspect packager may package the structural and non-structural aspects of the customer information data-entry fields selected at block 304 with selection box 402 of FIG. 4. In this illustrated embodiment, these aspects are packaged into a component, with the structural and non-structural aspects linked but separable.

[0044] This component comprises the structural aspect represented by the structural properties of the customer information fields (212, 214, 216, 218, 220, and 222 of FIG. 2), where they are arranged in the view of the existing electronic form template, and where they reside within the existing electronic form template's data structure.

[0045] This component also comprises non-structural aspects of the customer information fields. It comprises a

formatting aspect, here the font, size, and color of text in and around the data-entry fields, e.g., "Customer Information", "Car Type:", "Customer Name:", "Street:", "City:", "State:", and "ZIP Code:". The component is also built to comprise the view representation aspect, such as the shaded, dashed-line box marked at 224 in FIG. 2. It also comprises a data connection aspect for auto-populating the city and State fields based on a ZIP code.

[0046] At block 308, the aspect packager may create an icon or other graphic to aid in future graphical selection of the packaged aspect(s).

[0047] Continuing the illustrated embodiment, an exemplary view 502 of the component, entitled "customer information", is set forth in FIG. 5. This view of the component comprises an icon representing a smaller, simplified view of the portion of the existing electronic form template from which the component's aspects were packaged. This FIG. 5 also shows a design view 504 of current electronic form template 120, which is in the process of being built. The current electronic form template comprises a non-entry text box showing the title of the current electronic form template, "SERVICE RECORD", at 506.

[0048] Following block 306 and/or block 308, computer-readable media 108 of FIG. 1 may proceed (whether through the aspect packager, design application, or otherwise) to perform the acts of process 600 set forth below. In one embodiment it proceeds from process 300 to process 600 automatically, thereby adding the aspects selected in process 300 into current electronic form template 120 without further user or designer interaction.

Adding Designer-Created Aspects

[0049] The following discussion describes exemplary ways in which the tools enable an aspect to be added to an electronic form template.

[0050] An exemplary process 600 for enabling a designer to add a packaged designer-created aspect to an electronic form template is shown in FIG. 6. Process 600 is illustrated as a series of blocks representing individual operations or acts performed by elements of the operating environment 100 of FIG. 1, such as design application 116 and design user interface 118.

[0051] At block 602, the design application (and its user interface) enables selection of a packaged designer-created aspect. The design application may do so graphically or otherwise. In one embodiment the design application also enables selection of standard, built-in controls (not shown). In this embodiment, the standard, built-in controls may be selected and used similarly as the packaged, designer-created aspect.

[0052] Continuing the illustrated embodiment, the design application presents design view 504 of current electronic form template 120 shown in FIG. 5. The design application enables graphical selection of the packaged designer-created aspects represented by component 502.

[0053] At block 604, the design application receives a selection to add a packaged designer-created aspect to the current electronic form template. The design application may enable, and may receive, selection of a group of aspects together or singularly. The design application may receive a selection to add just the structural aspect, the structural

aspect and one or more of the non-structural aspects, or any number of the non-structural aspects without the structural aspect.

[0054] At block 606, if the designer selects to add the structural aspect or does not indicate whether or not to add the structural aspect, the design application proceeds along the “No” path to block 608. If the designer explicitly selects to add the structural aspect, the design application proceeds along the “Yes” path to block 610.

[0055] Continuing the illustrated embodiment, the designer drags the component (see 502 of FIG. 5) into an area occupied by the design view (not shown). In this case, the designer does not indicate whether or not he or she wishes to add the structural aspect. The design application proceeds to block 608.

[0056] At block 608, the design application determines whether or not the current electronic form template comprises a data structure similar to that of the structural aspect and/or to which a non-structural aspect may be mapped. If not, the design application proceeds along the “No” path to block 610. If yes, the design application proceeds along the “Yes” path to block 612.

[0057] The design application may make this determination using a “sniffer” program capable of analyzing the current electronic form template’s structure, such as a schema governing the current electronic form template. If the design application determines that the current electronic form template comprises such a data structure, it may map the non-structural aspect(s) to the similar data structure or ask the designer whether or not to do so and respond accordingly.

[0058] Continuing the illustrated embodiment, the design application determines that the current electronic form template being designed in FIG. 5 does not comprise a similar structure to that of the structural aspect. The design application then proceeds to block 610.

[0059] At block 610, the design application adds the structural aspect to the current electronic form template. In one embodiment, the design application alters the data structure of the current electronic form template to incorporate the structural aspect. It may do so by altering, for instance, a schema governing the current electronic form template.

[0060] Following or incident with this addition, the design application adds the selected non-structural aspects to the current electronic form template at block 614. In one embodiment, the design application adds the selected non-structural aspects by mapping these non-structural aspects to the added structural aspect and saving the non-structural aspects to files associated with the current electronic form template.

[0061] FIG. 7 shows this addition of the packaged designer-created aspects of the component to the current electronic form template at 702.

[0062] Returning to block 608, the design application may determine that the current electronic form template comprises a data structure similar to that of the structural aspect and/or to which a non-structural aspect may be mapped.

[0063] By way of example, FIG. 8 sets forth a second illustrated embodiment showing a design view 802 of a

current electronic form template 804 having an existing data structure. This current electronic form template is a standard form template designed for billing clients. Because the form template matches a standard (whether set by an industry, billing software company, or others), a form designer may not wish to alter its data structure, as that may make the form template’s intended use difficult or impossible. This form template comprises various controls, like data-entry fields and a selectable button.

[0064] At block 608, the design application determines that a set 806 of the data-entry fields (shown within a dashed-line box) has a data structure similar to that of the structural aspect of the customer information component shown in FIG. 8 at 502. This set may be governed by a schema similar to that of the structural aspect of the customer information component, for instance. This set may be similar in part due to its properties matching those of the customer information fields of FIG. 2. The set of fields 806 comprise, similar to the customer information fields of FIG. 2, six data-entry fields where two of them permit text or numbers, three permit only text, and one permits only numbers.

[0065] Responsive to this determination, the design application proceeds to block 612. At block 612, the design application maps selected non-structural aspects to a data structure of the current electronic form template. This data structure is similar to that of the structural aspect associated with the non-structural aspects and/or has a structure permitting mapping of the non-structural aspects. Through this mapping, the design application may enable a designer to add previously existing designer-created aspects to a current electronic form template that already has a data structure.

[0066] Continuing this illustrated embodiment, and assuming that the form designer selected all of the customer information component’s non-structural aspects, the design application maps each of the non-structural aspects to the similar data structure of the current electronic form template of FIG. 8.

[0067] The customer information component comprises, by way of review, three non-structural aspects shown in FIG. 1: data connection aspect 128; formatting aspect 134; and view representation aspect 136. The design application maps each 11 of these to the similar data structure of the current electronic form template and adds them to the current electronic form template at block 614.

[0068] This addition of the non-structural aspects is shown in part with FIG. 9. This figure presents a design-view representation of the additional formatting and view representation aspects. The data connection aspect for auto-populating the State and city fields is not shown. The formatting is shown with the addition of text having the font, size, and color also shown in FIG. 2 with “Customer Information” at 902, “Car Type:” at 904, “Customer Name:” and 906, “Street:” at 908, “City:” at 910, “State:” at 912, and “ZIP Code:” at 914. The view representation is shown with the shaded, dashed-line box marked at 916.

[0069] The structural or non-structural aspects enabled to be added to the current electronic form template, whether the form template has a similar data structure to that of a packaged designer-created aspect or not, may be added with information sufficient to permit the aspects to be viewed differently in a design view than a user view. Also, this information may enable an added designer-created aspect to be altered through the design view.

CONCLUSION

[0070] The above-described tools enable packaging and/or reuse of aspects for electronic form templates. Although the system and method has been described in language specific to structural features and/or methodological acts, it is to be understood that the system and method defined in the appended claims is not necessarily limited to the specific features or acts described. Rather, the specific features and acts are disclosed as exemplary forms of implementing the claimed system and method.

1. A method comprising:

enabling graphical selection of aspects originating from a first electronic form template;

receiving selection of the aspects; and

adding, responsive to the selection, the selected aspects to a second electronic form template.

2. The method of claim 1, wherein the act of receiving selection comprises receiving selection of less than all of the aspects enabled to be graphically selected.

3. The method of claim 2, wherein the act of adding comprises adding only the selected aspects.

4. The method of claim 1, wherein the act of enabling comprises enabling selection of non-structural aspects.

5. The method of claim 4, wherein the act of receiving comprises receiving selection of one of the non-structural aspects and further comprising mapping the non-structural aspect to an existing data structure of the second electronic form template if the existing data structure is similar to a structural aspect originating from the first electronic form template and associated with the non-structural aspect.

6. The method of claim 1, wherein: the act of enabling comprises enabling graphical selection of an aspect that determines formatting of a part of the first electronic form template; the act of receiving comprises receiving selection of this formatting aspect; and the act of adding comprises adding this formatting aspect to the second electronic form template effective to alter formatting of the second electronic form template.

7. The method of claim 1, wherein: the act of enabling comprises enabling graphical selection of an aspect comprising a data structure that governs at least part of the first electronic form template; the act of receiving comprises receiving selection of this data structure aspect; and the act of adding comprises adding the data structure to the second electronic form template.

8. A method comprising:

enabling graphical selection of one or more designer-created aspects of an electronic form template;

receiving selection of designer-created aspect(s) of the electronic form template; and

packaging the selected designer-created aspect(s) effective to enable the packaged aspect(s) to be added to another electronic form template.

9. The method of claim 8, wherein the act of enabling comprises enabling selection through a rendered view of the electronic form template.

10. The method of claim 8, wherein the act of enabling comprises enabling selection of all designer-created aspects of the electronic form template and the act of receiving selection comprises receiving selection of all of the designer-created aspects of the electronic form template.

11. The method of claim 8, wherein the act of packaging comprises packaging and saving the selected designer-created aspect(s) using an extensible markup language.

12. A method comprising:

enabling selection of a non-structural aspect created by a designer;

receiving selection of the non-structural aspect; and

mapping the selected non-structural aspect to a data structure of an electronic form template effective to add the selected non-structural aspect to the electronic form template.

13. The method of claim 12, wherein the selected non-structural aspect originates from another electronic form template.

14. The method of claim 12, wherein the act of enabling comprises enabling graphical selection of the non-structural aspect.

15. The method of claim 12, further comprising determining if the data structure of the electronic form template is one to which the selected non-structural aspect is capable of being mapped.

16. One or more computer-readable media having computer-readable instructions therein that, when executed by a computer, cause the computer to perform acts comprising:

enabling selection of designer-created aspects of a first electronic form template;

receiving selection of the designer-created aspects of the first electronic form template;

packaging the selected designer-created aspects for reuse; and

adding the packaged aspects to a second electronic form template.

17. The computer-readable media of claim 16, wherein the instructions cause the computer to perform the act of enabling to comprise enabling graphical selection of the designer-created aspects.

18. The computer-readable media of claim 16, wherein one of the packaged aspects is non-structural and wherein the instructions cause the computer to perform the act of adding to comprise mapping the non-structural packaged aspect to a data structure of the second electronic form template.

19. The computer-readable media of claim 18, wherein the instructions further cause the computer to perform acts comprising determining if the data structure of the second electronic form template is similar to a structural aspect of

the first electronic form template that is associated with the non-structural packaged aspect.

20. The computer-readable media of claim 16, wherein the instructions further cause the computer to perform acts comprising:

enabling graphical selection of the packaged aspects; and

receiving selection of the packaged aspects,

wherein the act of adding the packaged aspects is responsive to the act of receiving selection of the packaged aspects.

* * * * *